The state complexity of L^2 and L^k

Narad Rampersad
School of Computer Science
University of Waterloo
Waterloo, ON, N2L 3G1
CANADA
nrampersad@math.uwaterloo.ca

October 15, 2004

Abstract

We show that if M is a DFA with n states over an arbitrary alphabet and $L = L(M)$, then the worst-case state complexity of L^2 is $n2^n - 2^{n-1}$. If, however, M is a DFA over a unary alphabet, then the worst-case state complexity of L^k is $kn - k + 1$ for all $k \geq 2$.

1 Introduction

We are often interested in quantifying the complexity of a regular language L. One natural complexity measure for regular languages is the *state complexity* of L, that is, the number of states in the minimal deterministic finite automation (DFA) that accepts L. Given an operation on regular languages, we may also define the state complexity of that operation to be the number of states that are both sufficient and necessary in the worst-case for a DFA to accept the resulting language.

The first exact results for the state complexities of certain basic operations on regular languages such as concatenation, Kleene star, *etc.* were given by Yu, Zhuang, and Salomaa [7]. For instance, they proved that, given DFAs M_1 and M_2 with m and n states respectively, there exists a DFA with $m2^n - 2^{n-1}$ states that accepts $L(M_1)L(M_2)$. Moreover, there exist M_1 and M_2 for which this bound is optimal. Some more recent work on the state complexity of concatenation has been done by Jirásková [2] as well as Jirásek, Jirásková, and Szabari [3].

We are interested here in the state complexity of the concatenation of a regular language L with itself, which we denote L^2. We show that the bounds of Yu, Zhuang, and Salomaa for concatenation are also optimal for L^2. In other words, if M is a DFA with n states and $L = L(M)$, then the worst-case state complexity of L^2 is $n2^n - 2^{n-1}$. This bound, however, does not hold if we restrict ourselves to unary languages. Specifically, we show that if M is a DFA over a unary alphabet, then the worst-case state complexity of L^k is $kn - k + 1$ for all $k \geq 2$.

We first recall some basic definitions. For further details see [1]. A *deterministic finite automaton* M is a quintuple $M = (Q, \Sigma, \delta, q_0, F)$, where Q is a finite set of states; Σ is a finite alphabet; $\delta : Q \times \Sigma \to Q$ is the transition function, which we extend to $Q \times \Sigma^*$ in the natural way; $q_0 \in Q$ is the start state; and $F \subseteq Q$ is the set of final states. A DFA M accepts a word $w \in \Sigma^*$ if $\delta(q_0, w) \in F$. The language accepted by M is the set of all $w \in \Sigma^*$ such that $\delta(q_0, w) \in F$; this language is denoted $L(M)$. We denote the language $L(M)L(M)$ by $L^2(M)$. We may extend this notation to higher powers by the recursive definition $L^k(M) = L^{k-1}(M)L(M)$ for $k \geq 2$.

2 State complexity of L^2 for binary alphabets

In this section we consider the state complexity of L^2 for languages L over an alphabet of size at least 2.

Theorem 1. *For any integer $n \geq 3$, there exists a DFA M with n states such that the minimal DFA accepting the language $L^2(M)$ has $n2^n - 2^{n-1}$ states.*

Proof. That the minimal DFA for $L^2(M)$ has at most $n2^n - 2^{n-1}$ states follows from the upper bound of Yu, Zhuang, and Salomaa for concatenation of regular languages mentioned in the introduction. To show that $n2^n - 2^{n-1}$ states are also necessary in the worst case we define a DFA $M = (Q, \Sigma, \delta, 0, F)$ (Figure 1), where $Q = \{0, \ldots, n-1\}$, $\Sigma = \{0, 1\}$, $F = \{n-1\}$, and for any i, $0 \leq i \leq n-1$,

$$\delta(i, a) = \begin{cases} 0 & \text{if } a = 0 \text{ and } i = 1, \\ i & \text{if } a = 0 \text{ and } i \neq 1, \\ i + 1 \bmod n & \text{if } a = 1. \end{cases}$$

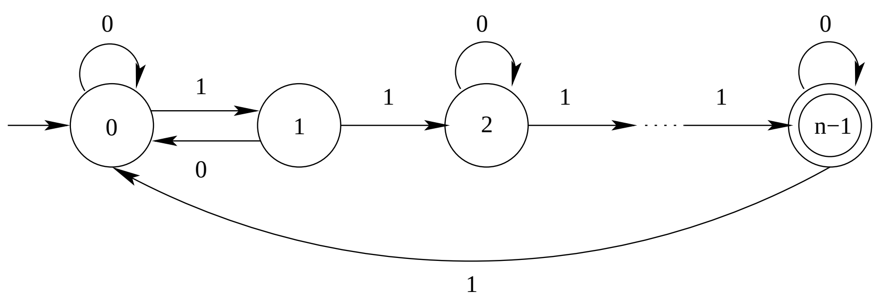

Figure 1: The DFA M

We will apply the construction of Yu, Zhuang, and Salomaa [7, Theorem 2.3] and show that the resulting DFA is minimal (see [3] for another example of this approach). Let $M' = (Q', \Sigma, \delta', (0, \emptyset), F')$, where

- $Q' = Q \times 2^Q - F \times 2^{Q-\{0\}}$;
- $F' = \{(q, R) \in Q' \mid R \cap F \neq \emptyset\}$; and
- $\delta'((q, R), a) = (\delta(q, a), R')$, for all $a \in \Sigma$, where

$$R' = \begin{cases} \delta(R, a) \cup \{\delta(0, a)\} & \text{if } q \in F, \\ \delta(R, a) & \text{otherwise.} \end{cases}$$

Then $L(M') = L^2(M)$ and M' has $n2^n - 2^{n-1}$ states.

To show that M' is minimal we will show (a) that all states of M' are reachable, and (b) that the states of M' are pairwise inequivalent with respect to the Myhill–Nerode equivalence relation [4, 6]. In what follows, all arithmetic is done modulo n.

To prove part (a) let (i, R) be a state of M', where $R = \{r_1, \ldots, r_k\}$. Let us assume that $0 \leq r_1 - 1 < \cdots < r_k - 1$. If $i = 0$, we see that

$$\delta'((0, \emptyset), 1^n(10)^{r_k - r_{k-1}} 1^n(10)^{r_{k-1} - r_{k-2}} \cdots 1^n(10)^{r_1 - 1}) = (0, R).$$

If $i > 0$, then let $R' = \{r_1 - i, \ldots, r_k - i\}$. Just as for $(0, R)$, we see that $(0, R')$ is reachable, and since $\delta'((0, R'), 1^i) = (i, R)$, (i, R) is also reachable, as required.

To prove part (b) let (i, R) and (j, S) be distinct states of M'. We have two cases.

Case 1: $R \neq S$. Then there exists r such that r is in one of R or S (say R) but not both. If $i \in F$, then $r \neq 0$. Hence $\delta'((i, R), 1^{n-1-r}) \in F'$ but $\delta'((j, S), 1^{n-1-r}) \notin F'$.

Case 2: $R = S$. Suppose $i - 1 < j - 1$. We have two subcases.

Case 2i: $j + 1 \notin S$. Then $\delta'((i, R), 1^{n-j}) = (i - j, R')$ for some R', and $\delta'((j, R), 1^{n-j}) = (0, S')$ for some S', where $1 \notin R'$ and $1 \in S'$. We may now apply the argument of Case 1 to the states $(i - j, R')$ and $(0, S')$.

Case 2ii: $j + 1 \in S$. Then let $k = 1$ if $i - j = 1$, and let $k = 0$ otherwise. Then $\delta'((i, R), 1^{n-j}1^k 0) = (i - j + k, R')$ for some R', and $\delta'((j, S), 1^{n-j}1^k 0) = (0, S')$, where either $R' \neq S'$ or $1 \notin R' = S'$. We may now apply the argument of either Case 1 or Case 2i, as appropriate, to the states $(i - j + k, R')$ and $(0, S')$. $\square$

3 State complexity of L^k for unary alphabets

In this section we show that the bound given in Theorem 1 does not hold if we restrict ourselves to unary languages. We also give optimal bounds for the state complexity of arbitrary powers L^k of a regular language L.

Following Pighizzini and Shallit [5], we note that the transition graph of a connected unary DFA M with n states is composed of a "tail" with $\mu \geq 0$ states and a "cycle" with $\lambda \geq 1$ states, where $n = \mu + \lambda$ (see [5, Figure 1]). We may therefore denote the size of M by the pair (λ, μ).

Pighizzini and Shallit give the following result regarding concatenation of unary DFAs.

Theorem 2 (Pighizzini and Shallit). *Let L_1, L_2 be unary languages accepted by DFAs of sizes $(\lambda_1, \mu_1), (\lambda_2, \mu_2)$ respectively. Then there exists a DFA M of size (λ, μ), where $\lambda = \operatorname{lcm}(\lambda_1, \lambda_2)$ and $\mu = \mu_1 + \mu_2 + \operatorname{lcm}(\lambda_1, \lambda_2) - 1$, such that $L(M) = L_1 L_2$.*

From Theorem 2 we can derive the following upper bound for the state complexity of L^k.

Theorem 3. *Let L be a unary language accepted by a DFA with n states. For all $k \geq 2$, there exists a DFA M with $kn - k + 1$ states such that $L(M) = L^k$.*

Proof. We prove the following by induction on k: if L is accepted by a DFA of size (λ, μ), where $n = \mu + \lambda$, then for all $k \geq 2$, there exists a DFA M of size $(\lambda, k\mu + (k-1)\lambda - k + 1)$ such that $L(M) = L^k$.

If $k = 2$, then an easy application of Theorem 2 with $L_1 = L_2 = L$ gives a DFA M of size $(\lambda, 2\mu + \lambda - 1)$ such that $L(M) = L^2$.

If $k > 2$, then write $L^k = L^{k-1}L$. By induction, L^{k-1} is accepted by a DFA of size $(\lambda, (k-1)\mu + (k-2)\lambda - k + 2)$. Applying Theorem 2 with $L_1 = L^{k-1}$ and $L_2 = L$ gives a DFA M of size $(\lambda, k\mu + (k-1)\lambda - k + 1)$ such that $L(M) = L^k$. The DFA M thus has

$$\begin{aligned} & \lambda + k\mu + (k-1)\lambda - k + 1 \\ = \; & k\mu + k\lambda - k + 1 \\ = \; & k(\mu + \lambda) - k + 1 \\ = \; & kn - k + 1 \end{aligned}$$

states, as required. $\square$

The following theorem gives a matching lower bound for the state complexity of L^k.

Theorem 4. *For any integers n, k, $n \geq 2$, $k \geq 2$, there exists a DFA M with n states over a unary alphabet such that the minimal DFA accepting the language $L^k(M)$ has $kn - k + 1$ states.*

Proof. We define a DFA $M = (Q, \Sigma, \delta, 0, F)$, where $Q = \{0, \ldots, n-1\}$, $\Sigma = \{0\}$, $F = \{n-1\}$, and for any i, $0 \leq i \leq n-1$, $\delta(i, 0) = i+1 \bmod n$. The transition graph of M is thus a directed n-cycle. Furthermore, $L(M) = 0^{n-1}(0^n)^*$. Hence, $L^k(M) = (0^{n-1}(0^n)^*)^k = 0^{k(n-1)}(0^n)^*$. The language $L^k(M)$ is accepted by the DFA $M' = (Q', \Sigma, \delta', 0, F')$, where $Q' = \{0, \ldots, kn - k\}$, $F' = \{kn-k\}$, for any i, $0 \leq i < kn-k$, $\delta'(i, 0) = i+1$, and $\delta'(kn-k, 0) = kn-k-n+1$. To see that the DFA M' is minimal, note that for any i, j, $0 \leq i < j \leq kn-k$, $\delta'(i, 0^{kn-k-j}) \notin F'$ and $\delta'(j, 0^{kn-k-j}) \in F'$. $\square$

4 Further work

It remains to investigate the worst-case state complexity of L^3, L^4, *etc.* for general alphabets.

5 Acknowledgements

Thanks to Jeffrey Shallit for suggesting this problem and for helpful discussions along the way.

References

[1] J.E. Hopcroft, J.D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley, 1979.

[2] G. Jirásková, "State complexity of some operations on regular languages". In proceedings of DCFS 2003, pp. 114–125, MTA SZTAKI, Budapest, 2003.

[3] J. Jirásek, G. Jirásková, A. Szabari, "State complexity of concatenation and complementation of regular languages". In pre-proceedings of CIAA 2004, pp. 132–142, Queen's University, Kingston, 2004.

[4] A. Nerode, "Linear automaton transformations", *Proc. Amer. Math. Soc.* **9** (1958), 541–544.

[5] G. Pighizzini, J. Shallit, "Unary language operations, state complexity and Jacobsthal's function", *Internat. J. Found. Comput. Sci.* **13** (2002), 145–159.

[6] M.O. Rabin, D. Scott, "Finite automata and their decision properties", *IBM J. Res. Develop.* **3** (1959), 114–125.

[7] S. Yu, Q. Zhuang, K. Salomaa, "The state complexities of some basic operations on regular languages", *Theoret. Comput. Sci.* **125** (1994), 315–328.